بسم الله الرحمن الرحيم

INTRODUCTION IN C++

ان شاء الله
المبرمج/أحمد عبد الرسول أمين



جمهورية مصر العربيه

محافظة المنيا-مركز مطاى

****

An Example C++ Program

Here is an example of a complete C++ program:

```
// The C++ compiler ignores comments which start with
// double slashes like this, up to the end of the line.
```

Some C++ compilers do not yet support name spaces

In this case you can use the older form of the include

directive (that does not require a using directive

and places all names

After the include and using directives, the basic structure of the program is:

```
#include <iostream.h>

int main()
{
first statement;
--------
--------
last statement;
return 0;
}
```

When at the end of the main program, the line

return 0;

means"return the value 0 to the computers operating system to signal that the program has completed succesfully"

Our example program uses four *variables* :

another_age and another_year ,age_now ,year_now

Program variables are not like variables in mathematics. They are more like symbol names for "pockets of computer memory" which can be used to store different values a different times during the program execution

Very Simple Input, Output and Assignment

After we have compiled the program above, we can run it .the result will be something like

.Enter current year then press RETURN

1996

Enter your current age in years

36

Enter the year for which you wish to know your

2001

Your age in 2001: 41

Will be you understanding after

The first, third, fifth and seventh lines above are produced on the screen by the program In general, the program statement

cout <<expression1<<expression2<<.......<<expression n;

will produce the screen output

Expression1Expression2........ExpressionN

The series of statements

```
Cout << expression 1 ;
Cout << expression 2 ;
.....
.....
Cout << expression n ;
```

will produce an identical output. If spaces or new lines are needed between the output expressions,these have to be included explicitly ,with a" " or a " \n" respectively. The expression endl can also be used to output a new line, and in many cases is preferable to using "n\" since it has the side-effect of flushing the output buffer (output is often stored internally and printed in chunks when sufficient output has

been accumulated;forces all output to appear on the screen endl using immediately).

The numbers in **bold in the example screen output above have been typed in** by, the user. In this The numbers in the program statement

```
cin >> year_now ;
```

has resulted in the variable the value 2001 *assigned* being year_now at the point when the user pressed RETURN after typing in "2001". Programs can also include assignment statements, a simple example of which is the statement

```
(another_age = another_year - (year_now - age_now;
```

Hence the symbol = means "is assigned the value of". ("Equals" represented in C++ as ==.)

Simple program :

Before looking at how to write C++ programs consider the following simple example Program.

```
// Sample program
// Reads values for thelength and width of a rectangle.
// and returns the perimeter and area of the rectangle.
#include <iostream.h>
Void main ()
{
int length, width;
int perimeter, area;
cout << "Length=";
Cin >> length;
Cout <<"width=";
Cin >>width ;
perimeter = 2*(length+width);
area = length*width;
Cout <<"perimeter is"<<perimeter<<endl ;
Cout <<"area is "<<area<<endl ; // out put results
}
// end of main program
```

سوف نتحدث عن:

(جمل اتخاذ القرار)

1- الجملة IF وتفرعاتها

2-جملة SWITCH

سوف نتعرف على الصيغة العامة لجملة IF

```
if (expression)
{
    statement1 ;

    statment2  ;
}
```

بامكاننا الاختصار الى القول انه اذا كان الشرط الذى تقوم الجملة IF باختياره صحيحا فقم بتنفيذ الجمل الى بين القوسين وفى حالة عدم الاختبار فلا تقوم بتنفيذ الجملة IF وانما استمر فى قراءة البرنامج من بعد كتلة IF

فمثلا انظر الى هذا الكود:

CODE

```
1- #include <iostream>
2- Using namespace std;
3- int main()
4- {
5- int i=0 ,j=0;
6- cin >> i >> j ;
7- if (i > j ) {
8- cout << "The number i is bigger than j" ;
9- }
10- return 0;
11- }
```

شرح البرنامج:

كما ترى ان هذا الكود يطلب من المستخدم ان يدخل رقمين,فيقوم البرنامج بمقارنة هذين الرقمين وفى حال الرقم الأول أكبر من الرقم الثانى فانه يطبع رسالة تخبرك بذلك وفى حال الرقم الثانى اكبروأو العددين متساويين فلن ينفذ السطر 8 لعدم صحة شرط جملة(IF).

مثال آخر:أكتب برنامج لاظهار على الشاشة x is positive

```
#include " iostream."
Using namespace std;
main ()
{
int x=5;
if (x>0)
cout<<x<<" x is positive";
return0;
}
```

جملة else/if :

لا يقوم الكود السابق بفعل اى شىء اذا أختل الشرط جملة if وبالرغم من اننا يمكننا أن نكتب جملة if أخرى وفى حال تساوى العددين نكتب جملة if ثالثة فلتفادى الأخطاء وللتسهيل نكتب else سوف نتعرف على الصيغه العامه لها

```
if (expression) {
 statement1 ;
 statement2;
}
else {
 statement3;
 statement4;
}
```

انه باختصار اذا لم يتحقق الشرط فى جملة if فان البرنامج سوف يقوم بتنفيذ الكتلة بعد العبارة else اما اذا تحقق الشرط فى جملة if فانه سينفذ الكتلة التى تتبع الجملة if وسوف يتجاهل الكتلة التى تتبع الجملة else .

الآن سنقوم باعادة الكود السابق و هذه المرة سنجعله يتعامل مع الحالات الآخرى .

CODE

```
#include <iostream>
using namespace std;
int main()
{
int i=0 ,j=0;
cin >> i >> j ;
if (i > j ) {
cout << "The number i is bigger than j" ;
}
else { cout << "error" ; }
```

```
22- return 0;
23- }
```

لم يختلف هذا البرنامج عن السابق الا فى جملة else سيقوم البرنامج بعرض العبارة خطا على الشاشة اذا لم يتحقق شرط فى عبارة if .

ممكن نطلق على عبارة if بانها ثنائيه لان البرنامج يتفرع الى فرعين أو طريقين اما عبارة ifالسابقه فانها احادية لان اذا لم يتحقق الشرط فتتجاهل عبارة if

فسوف نتحدث الآن عن عبارة if الثنائيه :

ممكن تصف بانها متعددة الاتجاهات و الصيغة العامة لها

```
If (expression) {
 statement1;
 statement2;
 statement3;
 }
else if (expression) {
 statment1;
 }
else if (expression) {
 statement;
 }
else {
 statement;
 }
```

سنقوم الآن بتطوير الكود السابق ليصبح قادرا على التعامل مع جميع الحالات .

CODE

```
1- #include <iostream>
2- using namespace std;
3- int main()
4- {
5- int i=0 ,j=0;
6- cin >> i >> j ;
7- if (i > j)  {
8- cout << "The number i is bigger than j" ;
9- }
10- else if (j > i) {
11- cout << "The number j is bigger than i" ;
12- }
13- else if ( j=i) {
14- cout << "there is no bigger number" ;
15- else { cout << "error" ; }
16- return 0;
17- }
```

نرى الختلاف مع الأكواد السابقه من السطر 10 الى 15 لاننا اضفنا لهذا الكود جماتين else if , تقوم الأولى ما اذا كان العدد الثانى أكبر وتطبع عبارة للمستخدم بذلك, تقوم الثانية باختبارما اذا كان العددان متساويان وتطبع

جملة للمستخدم بذلك , اما else الأخيرة فهى تفيدك فى وقوع أخطاء آخرى فتظهر على الشاشة للمستخدم (error)

مثال عملى:

سنقوم بكتابة برنامج شبيه ببرنامج الآله الحاسبه

CODE

```
#include <iostream>
using namespace std;

int main()
{
float a,b;
char x;

cout << "Enter Number1:\t" ;
cin >> a;
cout << "Enter Number2:\t" ;
cin >> b;
cout << "Enter the operator\t";
cin >> x;

cout << endl << endl;

cout << "Result:\t";


if (x=='+') { cout << a+b ;}
else if (x=='-') { cout << a-b;}
else if (x=='*') { cout << a*b;}
else if (x=='/') { cout << a/b;}
else { cout << "Bad Command";}

cout << endl;

return 0;
}
```

Soon the book will be Complete

سوف نتحدث عن :

(FOR LOOP) FOR الحلقة

نبذه عنها هى القيمه التى تحدد مرات التكرار

FOR (COUNTER STATEMENT; CONDITION; STEP)

COUNTER STATEMENT:وظيفة العداد لتسجيل عددالمرات التكرار
CONDTION:شرط الذى يحدد مرات التكرار ان يظل قائما او ينهييه
STEP:عدد مرات التكرار

EX:

```
# INCLUDE<IOSTREAM.H>
MAIN ()
{
INT COUNTER;
FOR (COUNTER=1; COUNTER<=10; COUNTER++)
COUT << COUNTER;
}
```

كان مثالا بسيطا للحلقه

من البرنامج السابق نجد ان الحلقه متبوعه بقوسين بينهما ثلاث عبارات تفصل بينهما الفصلة المنقوطه
العبارة الاولى:تخزن القيم الابتدائيه
العبارة الثانية:هى الشرط وهو يبدا من واحد وينتهى عند العشره
العباره الثالثة :يزداد العدد واحد كل مره فى الحلقه

ففى البرنامج السابق المخرجات هى اظهار الارقام من 1 الى 10